REMARKS

This amendment is responsive to the Final Office Action dated August 17, 2005. Applicants have amended claims 1, 5, 22, 23, and 30. Claims 1-30 remain pending.

Claim Rejection Under 35 U.S.C. § 103

In the Final Office Action, the Examiner rejected claims 1-9, 12-15 and 18-30 under 35 U.S.C. 103(a) as being unpatentable over Jardin (USPN 6,681,327) in view of Friedman et al. (USPN 6,240,513). The Examiner also rejected claims 10, 11, 16, 17 and 29 under 35 U.S.C. 103(a) as being unpatentable over Jardin in view of Friedman et al. as applied to claim 1 above, and in further view of Abramson et al. (USPN 6,539,494).

Applicants respectfully traverse the rejection. The applied references fail to disclose or suggest the inventions defined by Applicants' claims, and provide no teaching or suggestion of Applicants' claims

Claims 1-4, 6-7, 9-13, 15-21, 23-30

In relevant part, claim 1 is directed to a method in which an intermediary device performs the steps of (d) receiving encrypted application data from a client via a secure communications session; (e) decrypting encrypted application data; and (f) forwarding decrypted application data to the server via the secure network. Claim 1 also requires that the steps (e) and (f) are performed *at the packet level* of a network stack of the intermediate device *without processing the application data with an application layer of a network stack.* Thus, claim 1 requires that the intermediary device decrypt encrypted "application data" and forward the decrypted "application data" at the packet level without processing the application data with an application layer of a network stack.

In rejecting claim 1, the Examiner correctly recognizes that Jardin fails to teach or suggest these elements. However, the Examiner asserts that Friedman teaches a network security device that encrypts / decrypts and forwards packets at the packet level without reaching the network stack.

Applicants would like explain a subtle but fundamental difference between Applicants' claim 1 and the cited prior art. Applicants' acceleration device incorporates a "direct mode" in which the acceleration device provides accelerated decrypting and forwarding of *application*

PAGE 12/18 * RCVD AT 10/17/2005 4:11:06 PM [Eastern Daylight Time] * SVR:USPTO-EFXRF-6/30 * DNIS:2738300 * CSID:6517351102 * DURATION (mm-ss):04-12

data without requiring that the *application data* be reassembled at the application layer of the intermediate device. In other words, even though "application data" was encrypted at a client using conventional techniques, such as SSL, which operates above the packet level, the intermediate device is nevertheless able to decrypt and forward the application data at the packet level. This "direct mode" is described, for example, with respect to FIG. 5, pp. 14-18.

As explained in the present application (see, e.g., FIG. 5, block 265), in a client, the application layer protocol hands unencrypted application data to the session layer. At the session layer, the client uses SSL to encrypt the *application data* and hands the encrypted application data down through the layers to the network IP layer, where it is finally divided into packets and introduced into the physical layers. As a result, a single SSL security record sent by the client may constitute multiple packets. Normally, these "multi-segment packets" of encrypted application data would need to be passed up to the application-layer of the network stack to form a single SSL record for reassembly before the application data can be decrypted. However, in direct mode, Applicants' acceleration device decrypt the application data at the packet level without requiring that the packets be passed up the network stack. In this manner, the TCP layer of Applicants' acceleration device has been modified to support a direct mode in which the TCP layer itself provides certain additional functionality to bypass the session-layer and application layer of the network stack and directly decrypt application data that was encrypted at higher levels of the stack. As one example, pg. 15, ll. 17-27 of the present application describes certain features of the acceleration device that allow the upper levels of the network stack to be bypassed:

> *During decryption, the device may utilize portions of its memory to buffer segments as necessary for decryption. The number and size of the buffers will depend on the cipher scheme used and the configuration of the packets, as well as whether the packets contain <u>application data spanning multiple packets</u>, referred to herein as multi-segment packets (and illustrated with respect to Figure 8). The SSL device can allocate SSL buffers as necessary for TCP segments. If, for example, application data having a length of 3000 bytes is transmitted via TCP segments having a length of 100 bytes, the device can, copy TCP segment 1 to a first SSL buffer, and start a timer, wait for packet 2 and when received, copy it to an SSL buffer and restart the timer, and finally when packet 3 is received, the SSL accelerator will copy it, <u>decrypt all application data</u>, authenticate it and forward the data on in the clear (emphasis added).*

PAGE 13/18 * RCVD AT 10/17/2005 4:11:06 PM [Eastern Daylight Time] * SVR:USPTO-EFXRF-6/30 * DNIS:2738300 * CSID:6517351102 * DURATION (mm-ss):04-12

In contrast, neither Jardin nor Friedman teach decrypting "application data" at a packet level without processing the application data at an application layer, as required by claim 1. Friedman does not teach or suggest techniques by which an intermediate acceleration device receives encrypted "application data" and then decrypts that application data without processing the application data with an application layer.

To the contrary, Friedman describes a network device where both encryption and decryption are applied to individual packets for purposes of network security, i.e., at the packet level. In other words, the security device of Friedman is not receiving encrypted "application data" (i.e., application-layer data that was encrypted) and then decrypting the "application data." Rather, Friedman is only encrypting and decrypting packet-layer data, i.e., data specific to individual packets.

For example, in the Summary, Friedman states that "[i]t should be noted that encryption takes place at the <u>IP level</u> so that TCP and UDP packets are encoded" (emphasis added). Friedman further states that "[p]ackets received from the protected client are <u>encrypted</u> using an encipherment function such as IDEA, FEAL, or DES before being transmitted via the network to a destination" (emphasis added). Later in the Summary, Friedman describes how the "data and proprietary tail" of a given packet are encrypted before transmission of that particular packet on the network.

Thus, it is clear that Friedman is not concerned with decryption and forwarding of encrypted application data, i.e., application-layer data that was encrypted at a higher level of the network stack. Rather, both encryption and decryption are performed at the packet-level, i.e., the IP level. As a result, Friedman does not teach or suggest an intermediate acceleration device capable decrypting "application data" and forwarding the decrypted "application data" at the packet level without processing the application data with an application layer of a network stack. Friedman does not teach a solution to the problem addressed by Applicants' intermediate acceleration device, namely, an acceleration device capable of accelerating the processing of security records containing "application data." Friedmen fails to teach any mechanism capable of decrypting security records of application at a packet layer. For at least these reasons, Jardin in view of Friedman fails to teach or suggest an intermediate device that receives "encrypted application data," i.e., application data that has originally been encrypted at the application layer,

PAGE 14/18 * RCVD AT 10/17/2005 4:11:06 PM [Eastern Daylight Time] * SVR:USPTO-EFXRF-6/30 * DNIS:2738300 * CSID:6517351102 * DURATION (mm-ss):04-12

and decrypts and forwards the application data without processing the application data with an application layer of a network stack, as required by claim 1.

With respect to claim 12, Jardin in view of Friedman fails to teach or suggest an apparatus in which a proxy SSL communications engine and a server TCP communications engine decrypt *encrypted application data* from the client and forward the decrypted application data to the one of the plurality of servers without processing the application data with an application layer of a network stack of the apparatus.

With respect to claim 23, Applicants have amended claim 23 to clarify that the intermediate device receives communications directed to the enterprise in secure protocol, *wherein the secure protocol provides encrypted application data associated with an application layer of a network stack.* This claim amendment is consistent with other elements of the currently pending claims and, therefore, raises no new issues. Applicants request entry of this amendment.

Jardin in view of Friedman fails to teach or suggest receiving communications directed to the enterprise in secure protocol, wherein the secure protocol provides encrypted application data associated with an application layer of a network stack, and then decrypting data packets of the secure protocol to provide decrypted packet data at the packet-level of a network stack of the intermediate device. Further, Jardin fails to teach or suggest bypassing the application layer of the network stack of the intermediate device and forwarding the decrypted packet data from the intermediate device to at least one server of the enterprise without processing the decrypted packet data with the application layer.

Similarly, with respect to amended claim 30, Jardin in view of Friedman fails to teach or suggest bypassing an application layer of a network stack of the intermediate device and forwarding decrypted application data from the intermediate device to the server via the secure network without processing the decrypted application data with the application layer.

For at least these reasons, the rejection of claims 1, 12, 23 and 30 under 35 U.S.C. 103(a) should be withdrawn. Claims 2-4, 6-7, 9-11, 13, 15-21, 24-29 are patentable for at least the reasons set forth above with respect to the independent claims on which they depend.

PAGE 15/18 * RCVD AT 10/17/2005 4:11:06 PM [Eastern Daylight Time] * SVR:USPTO-EFXRF-6/30 * DNIS:2738300 * CSID:6517351102 * DURATION (mm-ss):04-12

Claims 5 and 22

In the Office Action, the Examiner recognized that Jardin fails to teach or suggest discarding at least a portion of each of the records after forwarding the portion to be discarded, and authenticating the decrypted application data of each data record using the remaining non-discarded portion of the data record upon receiving a final segment of the multi-segment record.

However, the Examiner asserted that Friedman teaches these elements. However, the cited portions of Friedman only describe the secure device dropping packets when the device has no key in its database to use to encrypt the packet. In other words, when no encryption key is designated for use with a particular destination device, all outbound packets from the client are dropped and are not securely forwarded to that device. This decision is made *prior* to sending any packets to the destination device. In Friedman, if a packet is discarded, it is simply not forwarded. Thus, contrary to the Examiner's assertion, Friedman does not teach or suggest discarding at least a portion of each of the records after forwarding those portions to the server, and then authenticating the security record using only the non-discarded portion, as required by claims 5 and 22.

Further, as described in detail above, Friedman clearly states that encryption / decryption and authentication occur on a per-packet basis. Thus, with respect to claims 5, 7 and 22, Friedman does not teach or suggest authenticating decrypted *application data* using a non-discarded portion of a multi-segment security record for that authentication data.

As a more general comment, Applicants' claims 5 and 22 are directed to a "bufferless or small buffer approach" described within pages 26 and 27 of the present application. In this embodiment, the intermediate acceleration device described by the Applicants uses "little or no buffer" when decrypting, forwarding and authenticating application data. For example, line 6 of page 26 first states that a "bufferless or small buffer approach" is used in one embodiment to handle a multisegment problem. In the bufferless and small buffer embodiments, the individual segments of an SSL record are <u>not buffered</u>, and authentication does not occur until the last segment of that SSL record is subsequently received. Jardin in view of Friedman fails to teach or suggest forwarding at least a portion of the decrypted application for each of the records prior to receiving complete records, discarding the portion of each of the records *after forwarding the portion to be discarded*, and then authenticating the decrypted application data of each data

PAGE 16/18 * RCVD AT 10/17/2005 4:11:06 PM [Eastern Daylight Time] * SVR:USPTO-EFXRF-6/30 * DNIS:2738300 * CSID:6517351102 * DURATION (mm-ss):04-12

record *using the remaining non-discarded portion* of the data record upon receiving a final segment of the multi-segment record.

For at least these reasons, the rejection of Applicants' claims 5 and 22 under 35 U.S.C. 103(a) should be withdrawn.

Claims 8 and 14

With respect to claim 8 and 14, the Examiner has failed to consider that Applicants' claims are directed to load balancing *sessions*. Applicants' claims 8 and 14 require prior to establishing a communications <u>session</u> with one of said plurality of servers, selecting one of said plurality of servers to forward the decrypted authentication data to based on a load balancing algorithm that calculates processing loads associated with each of the servers. Thus, Applicants' claimed acceleration devices controls the opening of new sessions based on currently calculated loads of the servers.

In contrast, Jardin describes a broker 130 that controls the flow of individual transactions to servers based on priorities associated with users. With respect to claims 8 and 14, the Examiner relied on column 8, lines 27-67 through col. 9, line 10 of Jardin. In the cited portions, however, Jardin first describes a "broker 130" capable of prioritizing transactions. Broker 130 monitors response times for individual transactions and then may elect to "reduce the flow of transactions" to one server over another. Controlling issuance of individual transactions is fundamentally different from actively load-balancing *sessions* across the servers by selecting the server based on current processing loads <u>prior</u> to establishing the communication session with the selected server, as required by Applicants' claims 8 and 14.

PAGE 17/18 * RCVD AT 10/17/2005 4:11:06 PM [Eastern Daylight Time] * SVR:USPTO-EFXRF-6/30 * DNIS:2738300 * CSID:6517351102 * DURATION (mm-ss):04-12

CONCLUSION

All claims in this application are in condition for allowance. Applicants respectfully request reconsideration and prompt allowance of all pending claims. Please charge any additional fees or credit any overpayment to deposit account number 50-1778. The Examiner is invited to telephone the below-signed attorney to discuss this application.

Date: October 17, 2005

SHUMAKER & SIEFFERT, P.A.
8425 Seasons Parkway, Suite 105
St. Paul, Minnesota 55125
Telephone: 651.735.1100
Facsimile: 651.735.1102

By: Kent J. Sieffert

Name: Kent J. Sieffert
Reg. No.: 41,312

PAGE 18/18 * RCVD AT 10/17/2005 4:11:06 PM [Eastern Daylight Time] * SVR:USPTO-EFXRF-6/30 * DNIS:2738300 * CSID:6517351102 * DURATION (mm-ss):04-12